# Programmierung und Bedienung des digitalen Kleinrechenautomaten C 8205

## Heft 4
## Bausteininterpretationssystem (BIS)

## Teil 1





Berichtigungen

Inhaltsverzeichnis

Punkt 8.2. : Matrizenmultiplikation

Abbildung

Punkt 1.2.1. : Der Dualstelle $z_o$ ist die Zweierpotenz $2^{31}$ zugeordnet.

Punkt 3.1.1. : Bei den jeweils vier Varianten für die Bil-
3.1.2. : dung und Übermittlung des Ergebnisses fehlen
3.2. : die Zeichen "⟨⟩" (Inhalt von).

Punkt 4. : Als Ansprungzellen für die Ausgabeprogramme werden die Zellen 7554 bis 7563 verwendet.

Punkt 4.4. : Tritt eine Pseudotetrade auf, so wird "P" ausgegeben.

Punkt 5.1. : Druck von ⟨7777⟩

Punkt 6.1.3. : Für den Speicherplatzbedarf der Rechenbausteine gilt im einzelnen:

| | | | |
|---|---|---|---|
| B 5.1. | $673_8$ | Speicherzellen | (14 Spuren) |
| B 5.2. | $115_8$ | Speicherzellen | ( 3 Spuren) |
| B 5.3. | $175_8$ | Speicherzellen | ( 4 Spuren) |
| B 5.4. | $177_8$ | Speicherzellen | ( 4 Spuren) |
| B 5.5. | $337_8$ | Speicherzellen | ( 7 Spuren) |

Punkt 6.1.5. : Für den Speicherbedarf der Servicebausteine gilt:

| | | | |
|---|---|---|---|
| B 6.1. | $533_8$ | Speicherzellen | (11 Spuren) |
| B 6.2. | $67_8$ | Speicherzellen | ( 2 Spuren) |
| B 6.3. | $37_8$ | Speicherzellen | ( 1 Spur ) |

Punkt 6.3.1. : (d) Ansprung des Hauptprogramms:

Punkt 6.3.6. : (a) Ansprung des KEP:
H - G - M↓ - E - E - M↑ - (T) - R

Punkt 7. : Bei der Eingabe mit Kennzeichen werden in beiden Codes alle nicht als erste Kennzeichen definierten Zeichen überlesen und zur

Eingabe eines neuen $K_1$ übergegangen.

**Punkt** 8.1. : Bei allen drei Beispielen fehlen in der
8.2. : Spalte "Bemerkungen" bei der Angabe des
8.3. : Inhaltes der Register $R_1$, $R_2$ und $R_3$ die Zeichen "$\langle\ \rangle$" (Inhalt von).

**Punkt** 8.1. : Ist $i^o > n^o$, so wird im Befehl 0022 zum UP Quadratwurzel gesprungen und aus dem in $R_1$ stehenden Quadrat des Betrages wird die Wurzel gezogen.

Inhaltsverzeichnis

## 0. Einleitung

Der interne Befehlsschlüssel des C 8205 läßt von den Grundrechenarten nur die Addition und Subtraktion von Festkommazahlen zu. Für einen sinnvollen Einsatz des Rechners in der Praxis ist jedoch Voraussetzung, daß man Festkommazahlen auch multiplizieren und dividieren kann.

Bei der Rechnung mit Festkommazahlen sind die Intervalle der zu verarbeitenden Zahlen stark eingeschränkt. Um die Anwendung des Automaten auch auf Gebiete ausdehnen zu können, bei denen sich die Größenordnung der Zahlen stark ändert, wurde die Gleitkommadarstellung geschaffen.

Da sich die verdrahtete Ein- und Ausgabe des Rechners nur auf ein Zeichen beziehen, müssen für die Ein- und Ausgabe von ganzen Worten ebenso wie für die Rechenoperationen mit Gleitkommazahlen spezielle Programme aufgestellt werden. Das Bausteininterpretiersystem besteht aus solchen Standardprogrammen für die Ein- und Ausgabe, für die Fest- und Gleitkommarechnung sowie Bedienungs- und Kontrollprogrammen, die für das Eingeben und Testen von Programmen benötigt werden. Gleichartige Programme sind zu sogenannten Bausteinen zusammengefaßt. Diese Bausteine können vom Benutzer entsprechend dem Bedarf zusammengesetzt werden.

Das Bausteininterpretiersystem enthält sowohl Programme für die Ein- und Ausgabe im Fernschreib-Code (5-Kanal-Code) als auch im R 300-Lochband-Code (8-Kanal-Code).

Bei der Ausarbeitung aller Bausteine - speziell bei den Programmen für die Fest- und Gleitkommaoperationen - wurde besonderer Wert auf Zeitoptimalität gelegt.

Alle Bausteine des Systems sind so aufgebaut,

daß sie auf sperrbaren Speicherplätzen abgespeichert und gegen Überschreiben gesichert werden können. Die Arbeitszellen der Programme befinden sich auf den letzten vier Spuren des Speichers. Das Protokollprogramm benutzt zusätzlich Speicherplätze der Spur 7500 - 7537 als Arbeitszellen.

Um die Programmierung zu erleichtern gibt es für den Ansprung der verschiedenen Ein- und Ausgabeprogramme sowie der Rechenprogramme feste Ansprungzellen, die auf der Spur 7540 - 7577 untergebracht sind.

## 1. Zahlwortinterpretation

Das interne Zahlwort stellt beim C 8205 eine Dualzahl dar, bei der das Komma an einer festen Stelle steht. Die Kommastellung kann im Prinzip durch den Benutzer gewählt werden. Sie ist bei der Programmierung von vornherein zu berücksichtigen. Grundsätzlich kann ein Zahlwort in beliebiger Weise interpretiert werden, auch als Zahl mit beweglichem Komma.

Beim Bausteininterpretiersystem wird das interne Zahlwort als Zahl mit festem Komma, als Gleitkommazahl oder als Zahl in Tetradendarstellung aufgefaßt. Für die interne Darstellung eines Zahlwortes stehen 33 Dualstellen zur Verfügung, die mit Null oder Eins besetzt sein können. Für diese 33 dualen Informationen gibt es eine der jeweiligen Bedeutung des Zahlwortes entsprechende kürzere externe Darstellung, die bei der Programmierung im allgemeinen verwandt wird. Die Umwandlung aus der externen in die interne Darstellung und umgekehrt erfolgt bei der Ein- bzw. Ausgabe durch den Automaten selbst. Bei der Behandlung der verschiedenen Zahlwortinterpretationen wird sowohl auf die interne als auch die externe Darstellung eingegangen.

### 1.1. Triadendarstellung

#### 1.1.1. Intern

Das Maschinenwort besteht, unabhängig von seiner speziellen Bedeutung, aus 33 Bits, die mit 0 bis 32 von links nach rechts durchnumeriert sind (Abb. 1). Jede dieser 33 Dualstellen ist mit einer der beiden Dualziffern Null und Eins besetzt. Insgesamt gibt es $2^{33}$ voneinander verschiedene Maschinenworte.

#### 1.1.2. Extern

Ohne Berücksichtigung der speziellen Bedeutung

eines Maschinenwortes, kann dieses durch eine externe Darstellung in oktaler Form beschrieben werden. Jeweils drei Dualziffern werden zu einer Oktalziffer (Triade) zusammengefaßt, so daß die den 33 Dualziffern entsprechende externe Darstellung aus 11 Oktalziffern besteht. Die Oktalziffern erhalten die Nummern 1 bis 11. Die Zuordnung der Wortstellen ist aus Abb. 2 ersichtlich. In Abb. 3 sind ein duales Maschinenwort und ein Wort in Triadendarstellung gegenübergestellt.

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 | 32 |

Abb. 1

| | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| 0 1 2 | 3 4 5 | 6 7 8 | 9 10 11 | 12 13 14 | 15 16 17 | 18 19 20 | 21 22 23 | 24 25 26 | 27 28 29 | 30 31 32 |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 |

Abb. 2

| | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| 100 | 111 | 001 | 101 | 000 | 011 | 110 | 010 | 100 | 101 | 011 |
| 4 | 7 | 1 | 5 | 0 | 3 | 6 | 2 | 4 | 5 | 3 |

Abb. 3

Die angeführte Triadendarstellung ermöglicht die externe Darstellung eines Maschinenwortes, die unabhängig von der Bedeutung desselben ist. Sie ist für jedes interne Wort anwendbar.

## 1.1. <u>Zahl in der Darstellung Festkomma hinten</u>

### 1.2.1. <u>Intern</u>

Man spricht von einer Festkommadarstellung, wenn jeder Ziffer einer Zahl ein fester Stellenwert zugeordnet wird. Beim C 8205 werden den in Abschnitt 1.1.1. angegebenen 33 Dualstellen die folgenden Stellenwerte zugeordnet (dual betrachtet):

70

| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | ... |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $2^{31}$ Vz | $2^{30}$ | $2^{29}$ | $2^{28}$ | $2^{27}$ | $2^{26}$ | $2^{25}$ | $2^{24}$ | $2^{23}$ | $2^{22}$ | $2^{21}$ | $2^{20}$ | $2^{19}$ | $2^{18}$ | $2^{17}$ | $2^{16}$ | $2^{15}$ | |

| ... | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 | 32 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | $2^{14}$ | $2^{13}$ | $2^{12}$ | $2^{11}$ | $2^{10}$ | $2^{9}$ | $2^{8}$ | $2^{7}$ | $2^{6}$ | $2^{5}$ | $2^{4}$ | $2^{3}$ | $2^{2}$ | $2^{1}$ | $2^{0}$ | - |

(nur für Rundungen od. V

Die Dualstelle $z_{32}$ wird nicht benutzt, so daß die der Stelle $z_i$ zugeordnete Zweierpotenz $2^{31-i}$ ist. Die Einheit der größten Stelle $z_o$ ist $2^{31}$, die der kleinsten $2^0$. Es können Zahlen aus dem Bereich

$$/z/ \leq 2^{31} - 1 = 2147\,483\,647$$

198 = 10

$z_{32}$ = Vorzeichen
0 = +
1 = −

intern durch die Maschine verarbeitet werden. (Die intern noch darstellbare Zahl $z = -2^{31}$ (nur $z_o$ besetzt) führt bei der Dezimalausgabe sowie der Multiplikation und Division zum Überlauf.)

Positive ganze Zahlen können in der Form

$$z = \sum_{i=1}^{31} z_i \cdot 2^{31-i}$$

geschrieben werden. Die kleinste von Null verschiedene Zahl ist $2^0 = 1$.

Negative Zahlen werden im Zweierkomplement dargestellt. Positive Zahlen sind durch $z_o = 0$, negative Zahlen durch $z_o = 1$ gekennzeichnet.

Das Komma steht am Ende des Wortes, d.h. zwischen der letzten durch die Zahl besetzten Dualstelle $z_{31}$ und $z_{32}$.

## 1.2.2. Extern

Die zugehörige Externdarstellung erfolgt dezimal. Die Konvertierung ins Dualsystem wird bei der Eingabe durch das Leseprogramm ausgeführt. Die externe Zahl besteht aus dem Vorzeichen und maximal 10 Dezimalziffern:

$$d = \overset{(+)}{-}\ d_9 d_8 d_7 d_6 d_5 d_4 d_3 d_2 d_1 d_0 \ .$$

Der externe Zahlenbereich ist durch

$$/d/ \leq 2^{31} - 1 = 2\ 147\ 483\ 647$$

beschränkt.

| | | |
|---|---|---|
| Beispiel: | internes Wort: | 00...001011010 |
| | zugehöriges externes Wort: | 45 |
| | externes Wort: | -45 |
| | zugehöriges internes Wort: | 11...110100110 |

Das zu -45 gehörende interne Wort erhält man durch Umrechnung der Zahl 45 ins Dualsystem, stellenweiser Negation, Addition einer Eins in der letzten Stelle und Linksverschiebung um eine Dualstelle.

## 1.3. Zahl in Gleitkommadarstellung

Jede Zahl kann in einer halblogarithmischen Schreibweise als Produkt aus Mantisse m und Basispotenz mit dem Exponenten e dargestellt werden:

$$z = \pm m \cdot B^e .$$

= GKZ (Gleitkommazahl)

Damit diese Darstellung eindeutig wird, muß für die Mantisse m die Normalisierungsvorschrift

$$B^{-1} \leq m < 1 \qquad (\text{für } z \neq 0)$$

gelten.

Die Mantisse m bestimmt die Ziffernfolge von z und der Exponent e legt fest, an welcher Stelle der Mantisse das Komma steht. B stellt die Basis des Zahlensystems dar.

### 1.3.1. Intern

Die interne Zahlendarstellung beruht auf der Schreibweise (Dual)

$$z = \pm m \cdot 2^e$$

Damit ergibt sich für die Mantisse die Normalisierungsvorschrift (dezimal betrachtet):

$$0{,}5 \leq m < 1 \qquad (\text{für } z \neq 0) \quad .$$

Exponent, Betrag der Mantisse und Vorzeichen werden separat gespeichert.

Der Exponent e umfaßt in der internen Darstellung die Stellen $z_1$ bis $z_6$. Für die Mantisse stehen die Stellen $z_7$ bis $z_{32}$ zur Verfügung.

Das Vorzeichen ist aus der Zahl herausgelöst und wird im Bit $z_0$ gespeichert.

Für die Vorzeichenstelle gilt:

$$\text{für } z \geq 0 \quad \text{ist} \quad z_0 = 0$$
$$\text{für } z < 0 \quad \text{ist} \quad z_0 = 1 \; .$$

Um Vorzeichenrechnungen im Exponenten zu vermeiden, wird der tatsächliche Exponent e intern um $2^5 = 32$ erhöht angegeben: $-32 \leq e \leq 31$

$$e' = e + 32 \; . \qquad 0 \leq e' \leq 63$$

Zwischen den einzelnen Dualstellen und der internen Gleitkommadarstellung ergibt sich somit ein Zusammenhang entsprechend Abb. 4.

| $z_0$ | $z_1$ | $z_2$ | $z_3$ | $z_4$ | $z_5$ | $z_6$ | $z_7$ | $z_8$ | $z_9$ | ... | $z_{30}$ | $z_{31}$ | $z_{32}$ |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| v | $2^5$ | $2^4$ | $2^3$ | $2^2$ | $2^1$ | $2^0$ | $2^{-1}$ | $2^{-2}$ | $2^{-3}$ | ... | $2^{-24}$ | $2^{-25}$ | $2^{-26}$ |

Vz    Exponent e'      Mantisse m

Die Gleitkommanull entspricht in der internen Darstellung der dualen Null, d.h. alle Dualstellen sind mit Null besetzt.

Die Normalisierungsvorschrift $0{,}5 \leq m < 1$ $(z \neq 0)$ besagt, daß bei allen Gleitkommazahlen, außer der Gleitkommanull, das Bit $z_7$ stets mit Eins besetzt sein muß.

Der tatsächliche Exponent e kann das Intervall

$$-32 \leq e \leq 31$$

durchlaufen und der Mantissenbereich ist durch

$$0{,}5 \leq m \leq 1-2^{-26} \qquad \text{(für alle } z \neq 0)$$

beschränkt.

Daraus ergibt sich der interne Zahlenbereich

$$/z/ \leq (1-2^{-26}) \cdot 2^{31} = 2^{31} - 2^{5}$$

Die kleinste darstellbare positive Zahl $\neq 0$ ist

$$z = 0{,}5 \cdot 2^{-32} = 2^{-33} \quad \leq /z/ \leq 2^{31}$$

### 1.3.2. Extern

Für die externe Darstellung einer Gleitkommazahl existieren zwei Möglichkeiten. Sie kann als Zahl in üblicher Dezimalschreibweise oder als Zahl in Gleitkommadarstellung angegeben werden, die sich sowohl für die Mantisse als auch den Exponenten auf das Dezimalsystem bezieht:

$$z = \pm m \cdot 10^{E}$$

Es gilt die Normalisierungsvorschrift:

$$0{,}1 \leq m < 1 \qquad \text{(für } z \neq 0)$$

Der Exponent E wird nicht erhöht. Er besteht aus dem Vorzeichen und dem Betrag von E

$$E = \pm /E/$$

Die externe Gleitkommazahl setzt sich zusammen aus dem Vorzeichen der Mantisse, maximal acht dezimalen Mantissenziffern, dem Exponentenvorzeichen und dem Exponenten E

$$z = \overset{(+)}{-}\, m_1 m_2 m_3 m_4 m_5 m_6 m_7 m_8 \overset{(+)}{-}\, E$$

Da der Exponent E bei der Eingabe nicht größer

als 8 sein darf, ist der Eingabezahlenbereich durch

$$/z/ \leq (1 - 10^{-8}) \cdot 10^{8}$$

beschränkt.

Die kleinste positive Zahl $\neq 0$, die eingegeben werden kann, ist

$$z = 0,1 \cdot 10^{-8} = 10^{-9}$$

Bei der externen Darstellung in Dezimalschreibweise wird die Gleitkommazahl in der Form

m = 26 Dual = 7 Dezim

$$z = \overset{(+)}{-} a, b$$

Nur 7 Dezimalziffern lassen sich genau verarbeiten

angegeben.

Der Eingabezahlenbereich ist wie bei der Gleitkommadarstellung durch

$$/z/ \leq (1 - 10^{-8}) \cdot 10^{8}$$

beschränkt. Die größte eingebbare Zahl ist 99 999 999, die kleinste $\neq 0$ ist 0,000000001.

## 1.4. Zahl in Tetradendarstellung

(Nur für Ein- u. Ausg

Bei der Tetradendarstellung von Dezimalzahlen wird jede Dezimalziffer in Form einer vierstelligen Dualzahl (Tetrade) geschrieben. Sie unterscheidet sich von der reinen Dezimaldarstellung lediglich dadurch, daß für die Beschreibung einer Ziffer nicht ein, sondern vier Zeichen verwendet werden. Bei der internen Darstellung können in den 33 Dualstellen des Maschinenwortes 8 Tetraden untergebracht werden. Die Zuordnung zwischen den in Abb. 1 angegebenen Wortstellen und den einzelnen Tetraden sieht folgendermaßen aus:

| $z_0$ $z_1$ $z_2$ $z_3$ | $z_4$ $z_5$ $z_6$ $z_7$ | $z_8$ $z_9$ $z_{10}$ $z_{11}$ | $z_{12}$ $z_{13}$ $z_{14}$ $z_{15}$ | ... |
|---|---|---|---|---|
| $t_7$ | $t_6$ | $t_5$ | $t_4$ | |

| ... | $z_{16}$ $z_{17}$ $z_{18}$ $z_{19}$ | $z_{20}$ $z_{21}$ $z_{22}$ $z_{23}$ | $z_{24}$ $z_{25}$ $z_{26}$ $z_{27}$ | $z_{28}$ $z_{29}$ $z_{30}$ $z_{31}$ | $z_{32}$ |
|---|---|---|---|---|---|
| | $t_3$ | $t_2$ | $t_1$ | $t_0$ | - |

Die Dualstelle $z_{32}$ wird wie bei der internen Festkommazahl nicht besetzt.

Bei der externen Darstellung besteht die Zahl aus maximal acht Dezimalziffern ohne Vorzeichen.

Der Zahlenbereich ist durch

$$z \leq 10^8 - 1$$ (Dezimal 0-9)

beschränkt.

Beispiel:

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| externes Wort | 2 | 8 | 1 | 3 | 6 | 9 | 0 | 5 | |
| internes Wort | 0010 | 1000 | 0001 | 0011 | 0110 | 1001 | 0000 | 0101 | 0 |

Im Bausteininterpretiersystem sind nur Programme für die Ein- und Ausgabe von Tetraden enthalten.

Rechenoperationen mit Zahlen in Tetradendarstellung können nicht durchgeführt werden.

## 2. Eingabe

Da beim C 8205 nur die Eingabe eines Zeichens fest verdrahtet ist, sind spezielle Leseprogramme erforderlich, um ganze Worte - z.B. Befehle oder Zahlen - eingeben zu können.

Das Bausteininterpretiersystem enthält Programme für die Eingabe im 5-Kanal-Fernschreib-Code (Internationales Telegraphenalphabet Nr. 2) und für die Eingabe im R 300-Code (8-Kanal-Code).

Die Eingabe kann über die Tastatur, die Schreibmaschine oder die beiden Lochbandleser erfolgen.

Die beim BIS verwendete Eingabeorganisation ermöglicht mit Hilfe von Eingabekennzeichen die Eingabe von Worten mit unterschiedlicher Bedeutung (Befehle, Zahlen in verschiedenartiger Darstellung). Außerdem sind im BIS Unterprogramme enthalten, die gestatten, Zahlen eines bestimmten Typs ohne die Angabe eines Kennzeichens einzulesen.

Die einzugebendenInformationen werden im allgemeinen durch eine Wortmarke "WM" abgeschlossen.

In der 5-Kanal-Version wird als Wortmarke das Zeichen Punkt "." benutzt.

In der 8-Kanal-Version wird bei Befehlen und Adressen das Zeichen Punkt (.) verwendet, während bei Zahlworten eines der beiden Zeichen Punkt (.) oder Tabulator (Tab) zur Markierung des Wortendes benutzt werden kann.

Bei allen Eingaben (Adressen, Befehle und Zahlen) können führende Nullen im Wort weggelassen werden.

Für die Eingabe von Zahlen in allen Darstellungen gilt, daß ein positives Vorzeichen nicht eingegeben werden darf, während ein negatives Vorzeichen an einer beliebigen Stelle auch innerhalb der Zahl vor der Wortmarke stehen kann.

## 2.1. Eingabe mit Kennzeichen

Um Befehle und Zahlen unterschiedlicher Darstellung eingeben zu können, muß es dem Automaten möglich sein, die ihm mitgeteilten Informationen zu unterscheiden. Außerdem müssen die eingelesenen Worte abgespeichert werden. – Befehle – Adressen – Zahlen

Zu diesem Zweck wurden für die Eingabe besondere Eingabekennzeichen ($K_1$ und $K_2$) und Hilfsadressen (Relativ-, Hauptleit-, Leitadressen) eingeführt. (f. Speicheradressen)

$K_1$, $K_2$, Information, WM

### 2.1.1. Relative Programmierung

Fast alle Programme werden relativ programmiert. Dadurch ist es möglich, ein aufgestelltes Programm auf einen beliebigen Speicherbereich einzugeben.

Bei der relativen Programmierung geht man bei der Numerierung eines Programms von der Adresse 0000 aus und Befehle, deren Adresse sich auf diese Anfangsadresse beziehen, werden markiert. Bei der Eingabe eines Programms wird der Adreßteil jedes so markierten Befehls um die Adresse erhöht, ab der die Abspeicherung beginnt. Diese Adresse wird beim C 8205 als Hauptleit- oder 1. Leitadresse bezeichnet.

Die Hauptleitadresse erfüllt zwei Funktionen. Sie bestimmt die Adresse, ab der das Programm gespeichert werden soll und dient zur Adressenänderung 1. Art.

Für die Adressenänderung 1. Art stehen beim BIS außerdem noch weitere 8 Adressen - sogenannte Leitadressen - zur Verfügung. Diese Adressen verwendet man z.B. als Anfangsadressen für Zahlenfelder, die vom Programm benötigt werden und zur Parameterversorgung von Unterprogrammen.

Im Gegensatz zur Hauptleitadresse werden die acht Leitadressen ausschließlich für die Adressenänderung 1. Art benutzt.

### 2.1.2. Eingabekennzeichen und deren Zuordnung

Zu jedem Befehl und jeder Zahl, die in den Automaten eingegeben werden sollen, gehört im allgemeinen ein "Erstes Kennzeichen $K_1$". Entsprechend diesem Kennzeichen $K_1$ wird das eingelesene Wort interpretiert und weiterverarbeitet.

Die Eingabekennzeichen werden nicht mit abgespeichert. Sie dienen nur zur Steuerung.

Als $K_1$ sind folgende Zeichen zugelassen:

Ziffern: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9
Buchstaben: D
Zeichen: Zwischenraum (ZwR), Schrägstrich (/)

Befehle haben für die Adressenänderung 1. Art ein Kennzeichen, das als "Zweites Kennzeichen $K_2$" bezeichnet wird. $K_2$ kann die Werte 0 bis 9 annehmen.

Alle definierten Kennzeichen $K_1$ und $K_2$ haben bei der Eingabe im Fernschreib-Code und im R 300-Code die gleiche Bedeutung.

$K_1 = 0$ Eingabe eines Maschinenwortes zur laufenden Speicherung

Nach der Eingabe des ersten und zweiten Kennzeichens können bis zu 11 Oktalziffern eingegeben werden:

$0K_2\ t_{10}t_9t_8t_7t_6t_5t_4t_3t_2t_1t_0$ .

Nach Eingabe der letzten Oktalziffer muß das Schlußzeichen Punkt "." folgen. Bei der Eingabe kann mit der ersten von Null verschiedenen Ziffer begonnen werden.

Für die Eingabe der Oktalzahl 0 wurde eine Sonderregelung getroffen. Nach Eingabe des ersten Kennzeichens $K_1 = 0$ kann sofort die Wortmarke "Punkt" gegeben werden:

0. (Null-Punkt)

Die eingelesenen Oktalziffern werden ohne Konvertierung abgespeichert und falls sie einen Internbefehl darstellen erst später abgearbeitet. Die Unterscheidung, ob das Wort als Befehl oder Zahl aufzufassen ist, hängt von seiner Verwendung ab. Es ist z.B. möglich, daß in einem Programm eine Oktalzahl sowohl als Befehl als auch als Konstante benutzt wird. In Abhängigkeit vom zweiten Kennzeichen $K_2$ kann vor der Abspeicherung eine Adressenänderung durchgeführt werden. Diese Änderung bezieht sich auf die Oktalstellen $t_3$ bis $t_{10}$, die damit den Adreßteil eines Maschinenwortes darstellen.

Das zweite Kennzeichen hat im einzelnen folgende Wirkung:

$K_2$ ="0" Das eingelesene Wort wird ohne Änderung der Oktalstellen $t_3$ bis $t_{10}$ abgespeichert. Falls das Wort einen Befehl darstellt, so ist die angegebene Adresse absolut.

$K_2$ ="1" Zu den Oktalstellen $t_3$ bis $t_{10}$ wird vor der Abspeicherung die Hauptleitadresse (= 1. Leitadresse) addiert (vgl. $K_1 = 2$, $K_2 = 1$).

$K_2$ ="2" Zu den Oktalstellen $t_3$ bis $t_{10}$ wird vor der Abspeicherung die 2. Leitadresse addiert (vgl. $K_1 = 2$, $K_2 = 2$).

⋮

$K_2$ ="9" Zu den Oktalstellen $t_3$ bis $t_{10}$ wird vor der Abspeicherung die 9. Leitadresse addiert (vgl. $K_1 = 2$, $K_2 = 9$)

$K_1$ = 1 Maschinenbefehl zur sofortigen Abarbeitung

Nach der Angabe des ersten und zweiten Kennzeichens könnten bis zu 11 Oktalzif-

fern eingegeben werden. Da das eingelesene Wort jedoch als Befehl interpretiert wird, ist nur die Eingabe von maximal 7 Ziffern sinnvoll.

1 $K_2$ $t_6t_5t_4t_3t_2t_1t_0$ .

Wie bei $K_1 = 0$ kann auch hier mit der ersten Oktalziffer ungleich Null begonnen werden und nach Eingabe der letzten Ziffer muß die Wortmarke Punkt (.) gegeben werden.

Das zweite Kennzeichen $K_2$ hat dieselbe Bedeutung wie bei $K_1 = 0$.

Ein Sofortbefehl wird nach seiner Eingabe in einer festen Zelle gespeichert und anschließend sofort abgearbeitet. Sofort-Sprungbefehle werden z.B. zum Anspringen von Programmen benötigt.

Wenn der Sofortbefehl kein Sprungbefehl ist, geht der Automat nach Ausführung des Befehls in den Zustand "unbedingter Stopp". Bei Betätigen der "Starttaste" (siehe Bedienungsanleitung) wird die Adresse des Sofortbefehls um eine Einheit erhöht, und der dadurch entstandene Befehl wird sofort ausgeführt. Anschliessend geht der Automat wieder in den Zustand "unbedingter Stopp". Mit Hilfe der Tastenkombination "H-G-R" kann zur Eingabe eines neuen ersten Kennzeichens zurückgekehrt werden.

So kann z.B. durch einen Sofort-Lesebefehl der Inhalt des durch die Adresse angegebenen Speicherplatzes in den Akkumulator gebracht und über ein Hilfsprogramm ausgegeben werden.

$K_1 = 2$ Hilfsadressen

Bei der Eingabe von Hilfsadressen können nach dem ersten und zweiten Kennzeichen maximal 8 Oktalziffern angegeben werden, die zwei oktale Adressen darstellen:

$2\ K_2\quad b_1b_2b_3b_4\quad a_1a_2a_3a_4$ .

Führende Nullen brauchen nicht eingegeben zu werden. Als Wortmarke wird das Zeichen Punkt "." benutzt. Die 8 Oktalziffern werden auf festen Speicherplätzen in den Oktalstellen $t_3$ bis $t_{10}$ gespeichert.

Die Möglichkeit Hilfsadressen mit 8 Ziffern eingeben zu können, bietet den Vorteil, daß auch Änderungen mit den ersten vier Stellen $t_{10}t_9t_8t_7$ einer Oktalzahl durchgeführt werden können (vgl. $K_1 = 0$).

Für das Kennzeichen $K_2$ sind die Ziffern 0 bis 9 zugelassen. Sie sind im einzelnen den folgenden Hilfsadressen zugeordnet:

$K_2$ ="0" Relativadresse

Bei der Relativadresse ist nur die Angabe der vier Oktalziffern $a_1a_2a_3a_4$ sinnvoll. Das Eingabebild hat die Gestalt:

$20\quad a_1a_2a_3a_4$ .

Mit Hilfe der Relativadresse kann die Abspeicherung in natürlicher Reihenfolge unterbrochen werden. Sie bewirkt einen Neuaufbau des Abspeicherbefehls (vgl. $K_1 = 2$, $K_2 = 1$). Die Abspeicherung wird ab der Adresse fortgesetzt, die sich aus der Summe von Hauptleit- und Relativadresse ergibt. Die Hauptleitadresse wird dabei nicht verändert, so daß sie weiterhin verwendet werden kann. Die Relativadresse steht danach nicht mehr zur Verfügung.

$K_2$ ="1" Hauptleitadresse (1. Leitadresse)

Die Hauptleitadresse kann mit maximal 8 Ziffern eingegeben werden.

21 $b_1b_2b_3b_4$ $a_1a_2a_3a_4$ .

Mittels der Hauptleitadresse wird der Abspeicherbefehl für die einzugebenden Worte aufgebaut. Die Speicherung beginnt bei der durch die Adresse $a_1a_2a_3a_4$ angegebenen Adresse. Die Adresse $b_1b_2b_3b_4$ hat keinen Einfluß auf die Abspeicherung eines Wortes. Nach jeder erfolgten Eingabe eines zu speichernden Wortes wird die Adresse des Abspeicherbefehls um eine Einheit erhöht, so daß eine Abspeicherung in natürlicher Reihenfolge gewährleistet ist. Die Speicherung in natürlicher Reihenfolge wird unterbrochen durch die Eingabe einer Relativadresse oder einer neuen Hauptleitadresse. Die Hauptleitadresse dient außerdem zur Adressenänderung 1. Art.

Durch die Angabe einer 8-stelligen Hauptleitadresse sind bei der Eingabe von Maschinenworten nicht nur Adressenänderungen mit der eigentlichen Adresse eines Befehlswortes ($t_6t_5t_4t_3$) möglich, sondern auch mit den Oktalstellen $t_{10}t_9t_8t_7$ (vgl. $K_1$ = 0).

Die Hauptleitadresse und der Speicherbefehl werden in festen Zellen gespeichert. Die Hauptleitadresse wird im allgemeinen erst durch die Eingabe einer neuen Hauptleitadresse zerstört.

$K_2$ ="2" 2. Leitadresse

$K_2$ ="3" 3. Leitadresse

⋮

$K_2$ ="9" 9. Leitadresse

Die zweite bis neunte Leitadresse können wie die Hauptleitadressen maximal 8 Oktalziffern umfassen. Das Eingabebild hat die Gestalt:

$$2\ K_2\ \ b_1 b_2 b_3 b_4\ \ a_1 a_2 a_3 a_4\ .$$

Die zweite bis neunte Leitadresse haben keinen Einfluß auf die laufende Speicherung von Worten. Sie werden ausschließlich zur Adressenänderung verwendet.

$K_1$ = 3 Dezimalzahl

Mit Hilfe des Kennzeichens $K_1$ = 3 kann eine Zahl in Dezimalschreibweise eingelesen werden. Das Eingabebild hat die Form:

3 (-) a, b WM bzw. 3 (-) a WM

Die Zahl wird intern in eine Gleitkommazahl umgewandelt. !

Der Eingabezahlenbereich ist durch

$/z/ \leq (1 - 10^{-8}) \cdot 10^8$ – 99 999 999

beschränkt (vgl. Abschnitt 1.3.2.).

Nach der ersten von Null verschiedenen Ziffer dürfen höchstens noch 7 Ziffern stehen.

Bei Zahlen mit $/z/ < 1$ dürfen zwischen Komma und erster von Null verschiedener Ziffer bis zu acht Nullen auftreten.

Die Zahl Null wird mit

3 0 WM

eingegeben.

$K_1$ = 4 Zahl in Gleitkommadarstellung

Eine Gleitkommazahl umfaßt maximal 14 Zeichen:

- erstes Kennzeichen
- negatives Vorzeichen der Mantisse
- maximal acht dezimale Mantissenziffern
- Mantissenschlußzeichen
- negatives Vorzeichen des Exponenten
- Exponent E
- Exponentenschlußzeichen

4 (-) $d_1d_2d_3d_4d_5d_6d_7d_8$ WM (-) E WM

Ein negatives Vorzeichen des Exponenten muß vor diesem eingegeben werden.

Alle Gleitkommazahlen mit Ausnahme der Null sind normalisiert einzugeben, d.h. in jedem Fall muß $d_1 \neq 0$ sein. Das Mantissenschlußzeichen kann nach der letzten Ziffer ungleich Null gegeben werden.

Die Null ist als Gleitkommazahl in der Form

4 0 WM

einzugeben.

Der Eingabezahlenbereich ist wie bei $K_1 = 3$ durch

$/z/ \leq (1 - 10^{-8}) \cdot 10^8$

beschränkt.

$K_1 = 5$ Zahl in der Darstellung Festkomma hinten

Es können ganze Zahlen aus dem Bereich

$/d/ \leq 2^{31} - 1 = 2\ 147\ 483\ 647$

eingegeben werden (vgl. Abschnitt 1.2.2.).

Diese bestehen aus maximal 13 Zeichen:

- erstes Kennzeichen
- negatives Vorzeichen
- maximal 10 Dezimalziffern
- Wortmarke

5 (-) $d_9d_8d_7d_6d_5d_4d_3d_2d_1d_0$ WM

Tritt ein Komma auf, so wird dieses überlesen. Die eingelesene Zahl wird in jedem Fall als ganzzahlig aufgefaßt.
Bemerkung: Die Kommastellung der einzugebenden Zahl ist vom Programmierer bzw. bei der Erstellung der Daten zu berücksichtigen. Alle Zahlen müssen gleiche Stellen hinter dem Komma haben.

$K_1$ = 6 Zahl in Tetradendarstellung

Es können ganze positive Zahlen aus dem Bereich

$d \leq 10^8 - 1$ = 99 999 999

eingelesen werden.

Das Eingabebild umfaßt maximal 10 Zeichen:

- erstes Kennzeichen
- maximal 8 Dezimalziffern
- Wortmarke

6 $d_7d_6d_5d_4d_3d_2d_1d_0$ WM

Ein auftretendes Komma wird überlesen.
Jede Dezimalziffer wird durch eine Tetrade dargestellt und in dieser Form abgespeichert.

$K_1$ = 7 Zahl in der Darstellung Festkomma hinten

Der Eingabezahlenbereich ist durch

$/d/ \leq 10^8 - 1$

beschränkt, Eine Zahl besteht aus maximal

11 Zeichen:

- erstes Kennzeichen
- negatives Vorzeichen
- maximal 8 Dezimalziffern
- Wortmarke

7 (-) $d_7 d_6 d_5 d_4 d_3 d_2 d_1 d_0$ WM

Wie bei $K_1 = 5$ und $K_1 = 6$ wird auch bei $K_1 = 7$ ein Komma überlesen und die Zahl als ganze Zahl angesehen.

Nach dem Einlesen und Darstellung jeder Dezimalziffer durch eine Tetrade wird die Zahl in eine Festkommazahl, Komma hinten konvertiert.

Mit Hilfe von $K_1 = 7$ ist eine schnellere Eingabe von Festkommazahlen, Komma hinten möglich als mit $K_1 = 5$.

$K_1 = 8$ Dezimalzahlen ohne Kennzeichen im Satz

Nach Eingabe dieses Kennzeichens $K_1 = 8$ können hintereinander Dezimalzahlen ohne erstes Kennzeichen $K_1 = 3$ eingelesen werden. Die Zahlen werden intern in Gleitkommazahlen konvertiert (vgl. $K_1 = 3$).

*Schlußzeichen = Satzmarke „:" „BM" „SM" hinter „WM" der letzten Zahl.*

$K_1 = 9$ Zahlen in der Darstellung Festkomma hinten ohne Kennzeichen im Satz

Nach $K_1 = 9$ können hintereinander Festkommazahlen, Komma hinten ohne erstes Kennzeichen $K_1 = 5$ eingelesen werden (vgl. $K_1 = 5$).

*$K_1 = 9$ ver[...] die Eingabe. Man locht $K_1 = 9$ → gibt alle Da[...] $K_1 = 5$ gilt nur für Einzeldaten*

Bei $K_1 = 8$ und $K_1 = 9$ können nur jeweils Zahlen gleichen Typs eingegeben werden. Die Abspeicherung erfolgt entsprechend der Hauptleitadresse, die vor dem ersten Kennzeichen $K_1 = 8$ oder $K_1 = 9$ anzugeben ist.

Bei der Eingabe im Fernschreib-Code wird ein Satz durch das Zeichen "Doppelpunkt (:)" abgeschlossen, während bei Verwendung des R 300-Codes die Satzeingabe durch die Zeichen "Satzmarke (SM)", "Blockmarke (BM)" und "Doppelpunkt (:)" beendet werden kann. Nach der Eingabe eines dieser Zeichen ist der Automat bereit, ein neues erstes Kennzeichen $K_1$ einzulesen.

$K_1$ = D Mit Hilfe dieses Kennzeichens werden Bedienungs- und Kontrollprogramme angesprungen. Ein zweites Kennzeichen bewirkt die Unterscheidung zwischen den einzelnen Programmen.

Die Beschreibung der Programme erfolgt in Abschnitt 5).

$K_1$ = <u>ZwR (Zwischenraum) Löschen</u>

Das erste Kennzeichen $K_1$ = ZwR wird zur Korrektur von Loch- und Tippfehlern und zur Gliederung des Lochbandes benutzt. Bei Eingabe von "ZwR" wird der Inhalt des Akkumulators gelöscht und zu einer neuen Eingabe übergegangen. Die Verwendung des Zeichens "ZwR" wird im Abschnitt 7 beschrieben.

$K_1$ = / <u>(Schrägstrich) Eingabe-Ende</u>

Dieses Zeichen steht im allgemeinen am Ende eines Lochbandes. Bei eingerastetem Schalter "BS" geht der Automat in bedingten Stopp. Durch Lösen von "BS" kann die Eingabe fortgesetzt werden.

## 2.1.3. Minimalprogramm

Das Minimalprogramm (MP) bildet den Grundbaustein des BIS. Es belegt die Speicherplätze 0 - 337.

Zum Minimalprogramm gehören neben anderen Programmen das Kleine Eingabeprogramm und ein Programm für die Eingabe mit Kennzeichen.

Von den als erste Kennzeichen definierten Zeichen verarbeitet das Minimalprogramm die Kennzeichen $K_1 = 0, 1, 2$ sowie $K_1$ = "ZwR" und $K_1$ = "/", d.h. im Minimalprogramm sind nur Leseprogramme für abzuspeichernde und für sofort auszuführende Maschinenworte, für Hilfsadressen sowie für die Zeichen "ZwR" und "Schrägstrich (/)" enthalten.

Für alle übrigen zugelassenen Kennzeichen sind im Minimalprogramm lediglich Anschlußbedingungen für die diesen Kennzeichen entsprechenden Eingabeprogramme vorhanden.

Wenn z.B. eine Dezimalzahl mit dem ersten Kennzeichen $K_1 = 3$ eingegeben werden soll, so muß vorher das Leseprogramm für Dezimalzahlen abgespeichert werden.

Bei Eingabe eines nicht als $K_1$ definierten Zeichens als erstes Kennzeichen wird - außer bei "Blockmarke" und ":" im R 300-Code und ":" im Fernschreib-Code - zur Eingabe eines neuen $K_1$ übergegangen.

## 2.1.4. Kleines Eingabeprogramm

Um bei einem Wechsel des Interpretationssystems oder beim Übergang vom 5- zum 8-Kanal-Code die meist recht umfangreichen Eingabeprogramme, wie z.B. die Minimalprogramme für die beiden Codes eingeben zu können, gibt es ein einfaches Leseprogramm. Dieses Programm bezeichnet man als "Kleines Eingabeprogramm (KEP)".

Das Kleine Eingabeprogramm steht auf der Bahn 0 und kann auf dem Speicherplatz 0002 angesprungen werden. Gegen Überschreiben ist es grundsätzlich gesichert (vgl. Bedienungsanleitung). Die Eingabe dieses Programms ist bei vollständig gelöschtem

Speicher nur über eine Hilfstastatur möglich.

Das Kleine Eingabeprogramm kann nur die fünf Ziffern und Zeichen "0", "1", Punkt ".", Doppelpunkt ":" und Zwischenraum "ZwR" verarbeiten. Es gibt kein Eingabekennzeichen.

Eine Adressenänderung ist nicht möglich. Alle Wörter müssen entsprechend ihrer internen dualen Darstellung durch Kombinationen von "0" und "1" eingegeben werden. Dabei kann mit dem ersten von Null verschiedenen Bit begonnen werden. Das Wort wird im Akkumulator aufgebaut. Als Schlußzeichen werden die beiden Zeichen Punkt und Doppelpunkt benutzt, während das Zeichen Zwischenraum zur Korrektur verwendet wird.

Durch das Schlußzeichen ":" wird die im Akkumulator aufgebaute Information als Befehl auf einem festen Speicherplatz gespeichert (Aufbau des Leitbefehles) und anschließend die Eingabe fortgesetzt.

Durch das Schlußzeichen "." wird der Leitbefehl ausgeführt. Falls er kein Sprungbefehl ist, wird anschließend seine Adresse um eine Einheit erhöht und zur Eingabe eines neuen Zeichens übergegangen.

Handelt es sich z.B. bei dem Leitbefehl um einen Transportbefehl $a_1a_2a_3a_4$ 240., so wird durch das Schlußzeichen Punkt das vorher im Akkumulator aufgebaute Wort nach der Zelle $a_1a_2a_3a_4$ abgespeichert. Die Erhöhung des Leitbefehles um eine Adresseneinheit gewährleistet die Abspeicherung in natürlicher Reihenfolge. Diese kann nur durch die Eingabe eines neuen Leitbefehls unterbrochen werden.

Soll der im Akkumulator aufgebaute Befehl sofort abgearbeitet werden, müssen als Schluß-

zeichen ":" und "." gegeben werden.

Durch das Zeichen "ZwR" können Tipp- und Lochfehler korrigiert werden, solange noch kein Schlußzeichen gegeben ist. Dabei wird der Akkumulatorinhalt gelöscht und das Wort kann nochmals eingegeben werden.

Befehl „025"

## 2.2. Eingabeunterprogramme

Bei der Eingabe mit Unterprogrammen wird von einem Hauptprogramm durch den Befehl "Sprung mit Rückkehrabsicht" zu einem Leseprogramm gesprungen, das als Unterprogramm aufgebaut ist. Nach dem Einlesen der Information und eines die Eingabe beendenden Schlußzeichens wird die Abarbeitung des Hauptprogrammes fortgesetzt. Die Eingabe kann über die Tastatur, das Schreibwerk oder die beiden Lochbandleser erfolgen.

### 2.2.1. Eingabeunterprogramme des Minimalprogramms

Im Minimalprogramm sind drei Eingabeprogramme enthalten, die als Unterprogramme verwendet werden können:

| | Ansprungzelle (Adresse) |
|---|---|
| Adresseneingabe | 316 |
| Eingabe eines Wortes mit Kennzeichen (K1, K2, Information) | 262 |
| Eingabe eines Programmstückes | 263 |

Nach dem Ansprung des Unterprogramms "Adresseneingabe" ist der Automat bereit eine maximal achtstellige oktale Adreßkonstante einzulesen. Dabei kann mit der ersten Ziffer ungleich Null begonnen werden. Als Schlußzeichen muß das Zeichen Punkt "." eingegeben werden. Bei einer Adreßkonstante Null ist nur die Eingabe des Schlußzeichens notwendig. Die Konstante steht nach dem Einlesen im Akkumulator und belegt die Wortstellen $z_0$ bis $z_{23}$. (8-stellige Oktalziffern) ($t_{10} - t_3$)

Bei Verwendung dieses Eingabeunterprogrammes darf das Zeichen "Zwischenraum" nicht gegeben werden, da "ZwR" einen Sprung zum Programm "Kennzeicheneingabe" bewirkt. In diesem Fall gibt es bei Angabe einer falschen Adresse keine Korrekturmöglichkeit.

Bei dem Unterprogramm "Eingabe eines Wortes mit Kennzeichen" erfolgt die Interpretation der einzuge-

benden Informationen entsprechend der in Abschnitt 2.1.2. beschriebenen Eingabekennzeichen. Es wird nur ein Wort in den Akkumulator eingelesen. Die Abspeicherung oder Weiterverarbeitung muß durch das Hauptprogramm erfolgen. Bei Eingabe einer Relativ- oder Hauptleitadresse mit 20 bzw. 21 wird nicht zum Hauptprogramm zurückgekehrt. Der Automat wartet dann auf die Eingabe eines neuen ersten Kennzeichens. Wird eine Leitadresse (2. bis 9.) eingegeben, so liest der Rechner anschließend ein weiteres Wort mit Kennzeichen ein, ehe zum Hauptprogramm zurückgekehrt wird.

Bei Verwendung des Unterprogramms "Eingabe eines Programmstückes" wird die Regie dem im Minimalprogramm enthaltenen Programm für die Kennzeicheneingabe übergeben. Es können ganze Programme oder auch Datenblöcke eingegeben werden. Die zugehörige Eingabeorganisation wurde in Abschnitt 2.1. erläutert. Die Abspeicherung des Programmstückes erfolgt ab der durch die Hauptleitadresse angegebenen Speicherzelle. Die Hauptleitadresse kann über die Tastatur eingegeben werden, oder sie kann auf dem Lochband stehen.

- Einlesen von Programmen
- Angeben, wohin das Befehlsprogr. eingelesen werden soll.

Die Rückkehr ins Hauptprogramm ist durch Eingabe des Schlußzeichens möglich. Als Schlußzeichen werden im R 300-Code die Zeichen Blockmarke "BM" und Doppelpunkt ":" und im Fernschreib-Code das Zeichen Doppelpunkt verwendet.

Zur Korrektur einer falschen Adresse darf nicht ZR benutzt werden

Will man an eine andere Stelle des Hauptprogrammes zurückkehren, so kann als Schlußzeichen ein entsprechender Sofortbefehl gegeben werden.

### 2.2.2. Eingabe einer Zahl ohne Kennzeichen

Um bei der Benutzung der Eingabeunterprogramme sowie aller nicht im Minimalprogramm enthaltenen Programme des BIS dem Programmierer die Arbeit zu erleichtern, sind für den Ansprung der Unterprogramme bestimmte Speicherzellen festgelegt worden. Diese Speicherzellen befinden sich auf der Spur 7540. Durch die Benutzung von festen Ansprungzellen wird umgangen, daß

sich der Programmierer bei der Eingabe eines Programmes auf verschiedene Speicherplätze auch die sich ändernden Ansprungadressen zu merken braucht.

Für das Einlesen einer Zahl ohne Kennzeichen existieren 6 verschiedene Eingabemöglichkeiten:

| | Ansprungzelle |
|---|---|
| (a) Eingabe einer Zahl in Gleitkommadarstellung (vgl. $K_1$=4) | 7575 |
| (b) Eingabe einer Zahl in Dezimalschreibweise (intern Umwandlung in eine Gleitkommazahl; vgl. $K_1$=3) | 7574 |
| (c) Eingabe einer Zahl in Dezimalschreibweise (intern Umwandlung in eine Festkommazahl +, K.h.; vgl. $K_1$ = 5) | 7573 |
| (d) Eingabe einer Zahl in Dezimalschreibweise mit Kommainformation (intern Umwandlung in eine Festkommazahl, K.h.) | 7572 |
| (e) Eingabe einer Zahl in Dezimalschreibweise mit maximal 8 Ziffern (intern Umwandlung in eine Festkommazahl, K.h.; vgl. $K_1$=7) | 7571 |
| (f) Eingabe einer maximal 8-stelligen Zahl in Dezimalschreibweise ohne Vorzeichen ≥0 (intern Umwandlung in eine Zahl in Tetradendarstellung; vgl. $K_1$=6) | 7570 |

Bei Verwendung dieser Eingabeprogramme steht die Zahl nach dem Einlesen im Akkumulator. Die Weiterverarbeitung muß durch das Hauptprogramm erfolgen.

Die unter (d) aufgeführte Eingabemöglichkeit gibt es bei der Eingabe mit Kennzeichen nicht. Bei der Eingabe einer Zahl mittels dieses Leseprogramms ist eine Kommainformation notwendig. Es muß die Anzahl der Stellen nach dem Komma angegeben werden, mit welcher die Zahl eingelesen werden soll. Dieser Parameter m ($0 \leq m \leq 15$) steht in den Bits $z_8$ $z_9$ $z_{10}$ $z_{11}$ des Befehls "Sprung mit Rückkehrabsicht nach 7572". (Die Bits $z_0$ bis $z_{11}$ sind bei einem Maschinenbefehl im allgemeinen nicht besetzt.) Der Parameter m bewirkt,

allg.: XXXX 7572 025
m=8 XX10 7572 025
m=6 XX06 7572 025

daß die Zahl beim Einlesen mit $10^m$ multipliziert wird, auch wenn ihre Stellenzahl hinter dem Komma kleiner als m ist. Für den Eingabezahlenbereich und die einzugebenden Zeichen gilt das bei $K_1=5$ angegebene mit den Außnahmen, daß kein erstes Kennzeichen $K_1=5$ auftreten darf und daß ein Komma nicht überlesen wird.

Beispiel für die Wirkung der Kommainformation:

31,74 bei m=0 ergibt: 31.

Als Kommainformation ist m=5 angegeben, d.h. die Zahl soll mit fünf Stellen nach dem Komma eingelesen werden. Der Eingabebefehl hat die Form 00 5 7572 025.. Die einzugebende Zahl ist 31,74, d.h. sie besitzt nur zwei Stellen nach dem Komma. Während der Eingabe wird die Zahl mit $10^m=10^5$ multipliziert, so daß danach die Zahl

3174000

im Akkumulator steht.

## 2.2.3. Satzeingabe von Zahlen ohne Kennzeichen

Auch die Unterprogramme für die Zahleneingabe im Satz ohne Kennzeichen sind über feste Ansprungzellen erreichbar. Die folgenden vier Eingabeprogramme können auf den Zellen 7564 bis 7567 angesprungen werden.

| | Ansprungzelle |
|---|---|
| (a) Eingabe von Zahlen in Gleitkommadarstellung im Satz (vergl. $K_1=4$) | 7567 |
| (b) Eingabe von Zahlen in Dezimalschreibweise im Satz (vgl. $K_1=8$)<br>(Die eingegebenen Zahlen werden intern in Gleitkommazahlen umgewandelt) | 7566 |
| (c) Eingabe von Zahlen in Dezimalschreibweise im Satz (vgl. $K_1=9$)<br>(Die eingelesenen Zahlen werden in Festkommazahlen, K.h. konvertiert) gleiche Kommastellen beachten! | 7565 |
| (d) Eingabe von Zahlen in Dezimalschreibweise mit Kommainformation im Satz (vgl. 2.2.2. (d))<br>(Die eingegebenen Zahlen werden in Festkommazahlen, K.h. umgewandelt.) | 7564 |

Kommastellung bestimmt der Automat wieder

den in Festkommazahlen, K.h. umgewandelt)

Vor dem Ansprung dieser Eingabeprogramme muß der Abspeicherbefehl durch das Hauptprogramm in der Zelle 7745 aufgebaut werden:

$$7745 \quad a_1a_2a_3a_4 \quad 240.$$

Dabei gibt $a_1a_2a_3a_4$ die Adresse an, ab der die Speicherung beginnen soll.

Die Rückkehr ins Hauptprogramm ist durch das Schlußzeichen möglich. Im Fernschreib-Code wird als Schlußzeichen eines Satzes das Zeichen Doppelpunkt ":" verwendet. Im R 300-Code kann ein Satz sowohl durch das Zeichen"Satzmarke/WRZL" als auch durch das Zeichen "Blockmarke (BM)" abgeschlossen werden. Ein Unterschied ergibt sich aus der Wirkung der beiden Zeichen.

Wenn man annimmt, daß der Befehl "Sprung mit Rückkehrabsicht zur Satzeingabe" in der Zelle n des Hauptprogrammes steht, so wird bei Eingabe von "SM" zur Zelle n+1 zurückgesprungen, während bei Eingabe von "BM" die Rückkehr zur Zelle n+2 erfolgt. Aus dieser Festlegung ergibt sich, daß nur eine Satzeingabe möglich ist.

In allen im Abschnitt 2.2. behandelten Eingabeunterprogrammen sind keine Anwahlbefehle berücksichtigt.

### 2.2.4. Bemerkungen zur Eingabe

Bei Satzeingabe im R 300-Code hat das Zeichen Doppelpunkt ":" dieselbe Wirkung wie das Zeichen Blockmarke "BM".

Die Satzschlußzeichen "SM/WRZL", "BM" und ":" schliessen die Funktion der Wortmarke nicht ein, d.h. vor der Satzmarke ist die Marke des letzten Wortes anzugeben.

Ein Wort und ein Satz können stets gelöscht werden, solange nicht die entsprechenden Schlußzeichen ein-

gegeben sind.

Das Löschen einer Zahl, eines Befehls oder einer Adresse kann durch das Zeichen Zwischenraum "ZWR" geschehen, während das Löschen eines Satzes in der 8-Kanal-Version durch die Zeichen "Irrung Satz (I-S)", "Irrung Block (I-B)" und Apostroph "'" und in der 5-Kanal-Version durch das Zeichen "H" erreicht werden kann. Das Irrungszeichen kann an einer beliebigen Stelle des Satzes vor dem Satzschlußzeichen gegeben werden. Nach dem Löschen eines Wortes oder eines Satzes kann das Wort bzw. der Satz neu eingegeben werden. Die erneute Satzeingabe ist in jedem Fall auch bei der Eingabe mit Kennzeichen $K_1=8$ oder $K_1=9$ mit der ersten Zahl zu beginnen, während ein Wort bei der Eingabe mit Kennzeichen beginnend mit dem ersten Kennzeichen neu eingegeben werden muß.

Kein direktes Löschen, sondern Überlesen des falschen Textes

Im 5-Kanal-Code hat das Auftreten der Zeichen Wagenrücklauf "WR", Zeilenvorschub "ZV", Ziffern/Zeichen "Zi/Zei" und Buchstaben "Bu" zwischen zwei Zahlen keine Wirkung. Dasselbe gilt im 8-Kanal-Code nur für die Zeichen "WR", Großbuchstaben "GB" und Kleinbuchstaben "KB". Tritt hier "ZV" auf, so ist anschließend mit "ZwR" zu löschen.

## 3. Rechenoperationen

Da beim Cellatron C 8205 an Rechenoperationen nur die Festkommaaddition bzw. -subtraktion verdrahtet sind, müssen die Festkommamultiplikation und -division sowie alle Gleitkommarechenoperationen über Unterprogramme realisiert werden. Diese Standardprogramme für die Festkommamultiplikation und -division, die Gleitkommaaddition und -subtraktion, die Gleitkommamultiplikation und -division sind im BIS als Bausteine enthalten. Die Parameterversorgung für diese sogenannten "offenen Unterprogramme" weicht ab von den Vereinbarungen über geschlossene Unterprogramme (vgl. Anleitung zur Programmierung in der Maschinensprache). Sie wird besonders geregelt. Da der C 8205 außer dem Akkumulator im Rechenwerk kein weiteres Register besitzt, sind für die Parameterversorgung der offenen Unterprogramme verschiedene "Register" definiert worden. Es handelt sich dabei um drei Speicherzellen auf der letzten Spur und zwar:

Register $R_1$: Zelle 7773
Register $R_2$: Zelle 7776
Register $R_3$: Zelle 7774

Ihre Anwendung bei den einzelnen Programmen ist an bestimmte Regeln gebunden, die vom Programmierer beachtet werden müssen.

Vor dem Aufruf eines Unterprogrammes sind die Operanden durch das Hauptprogramm in die erwähnten Register abzuspeichern. Der Aufruf des Unterprogrammes braucht nicht direkt nach der Speicherung der Operanden zu erfolgen, wenn man beachtet, daß die Inhalte der entsprechenden Register inzwischen nicht zerstört werden.

Damit der Programmierer sich bei Abspeicherung der Bausteine auf verschiedene Speicherplätze nicht jeweils neue Einsprungadressen zu merken braucht, wurden auch hier Ansprungzellen festgelegt. Es werden dazu die Zellen 7540 bis 7553 verwendet. Für die Er-

gebnisübermittlung gibt es grundsätzlich zwei Möglichkeiten:

- das Ergebnis wird dem Hauptprogramm nur im Akkumulator übermittelt
- das Ergebnis wird dem Hauptprogramm im Akkumulator und im Register $R_1$ übermittelt

Wird bei den Gleit- oder Festkommarechnungen der interne Zahlenbereich überschritten, so wird im jeweiligen Unterprogramm der Zustand Überlauf erzeugt und der Rechner geht in den Zustand "unbedingter Stopp". Der Adreßteil des unbedingten Stoppbefehls, der mit Hilfe der Befehlsregisteranzeige kontrolliert werden kann, gibt Auskunft, in welchem Rechenprogramm die Überschreitung des Zahlenbereiches eingetreten ist (vgl. Bedienungsanleitung). In der Adressenanzeige wird die Ansprungzelle des Programms angegeben. Wird z.B. bei der Multiplikation zweier Gleitkommazahlen das Produkt größer als $z = (1-2^{-26}).2^{31}$, so wird die Ansprungadresse des Rechenprogrammes "Gleitkommamultiplikation" - 7546 - angezeigt.

Will man sehen, welcher Unterprogramm-Aufruf des Hauptprogramms abgearbeitet wurde, kann man mit Hilfe eines Programmes (z.B. Bereichsausschreiben/D-3) die gespeicherten Rücksprungadressen ausschreiben.

Bei Unterschreitung des Wertebereiches wird das Ergebnis zu Null definiert.

## 3.1. Gleitkommaoperationen

### 3.1.1. Addition, Subtraktion

Die Gleitkommaaddition und -subtraktion benutzen als Operandenregister $R_1$ und $R_2$.

Die bei der Addition und Subtraktion entsprechend der Ergebnisübermittlung möglichen vier Varianten können über die folgenden Ansprungzellen erreicht werden:

| | Ansprungzelle |
|---|---|
| (a) Gleitkommaaddition mit Speicherung | 7553 |

| | | Ansprungzelle |
|---|---|---|
| (b) | Gleitkommaaddition ohne Speicherung | 7552 |
| (c) | Gleitkommasubtraktion mit Speicherung | 7551 |
| (d) | Gleitkommasubtraktion ohne Speicherung | 7550 |

Bei den vier Varianten wird im einzelnen gebildet:

t = 30 - 40 msec

(a) $\langle R_1 \rangle := \langle Ac \rangle := \langle R_1 \rangle + \langle R_2 \rangle$ 7553

(b) $\langle Ac \rangle := \langle R_1 \rangle + \langle R_2 \rangle$ 7552

(c) $\langle R_1 \rangle := \langle Ac \rangle := \langle R_2 \rangle - \langle R_1 \rangle$ 7551

(d) $\langle Ac \rangle := \langle R_2 \rangle - \langle R_1 \rangle$ 7550

(a) u. (c): Ausgangswert bleibt nur in $\langle R_2 \rangle$ denn $\langle R_1 \rangle$ wird zu Ergebnis

Bei der Subtraktion wird so verfahren, daß zunächst der Inhalt von $R_1$ mit (-1) multipliziert wird und danach $R_1$ und $R_2$ addiert werden.

Nach der Rechnung stehen im Fall:

(a) und (c) in $R_2$ der Ausgangswert, in $R_1$ das Ergebnis;
(b) in $R_1$ und $R_2$ die Ausgangswerte;
(d) in $R_2$ der Ausgangswert, in $R_1$ der mit (-1) multiplizierte Ausgangswert.

$(1-2^{-26}) \cdot 2^{31}$

Bei Überschreitung des Zahlenbereiches leuchten die Lampen "FÜ" und "US" und in der Befehlsregisteranzeige wird ein unbedingter Stoppbefehl mit der Adresse 7553 angegeben.

FÜ = Fehler - Überlauf
US = unbedingter Stopp

Die Rücksprungadresse ist für die

- Addition und Subtraktion ohne Speicherung in der Zelle 7707
- Addition und Subtraktion mit Speicherung in der Zelle 7710

gespeichert.

Bei den Programmen mit Speicherung steht in der Zelle 7707 der Befehl "7773 240" (Transport nach Zelle 7773).

### 3.1.2. Multiplikation, Division

Die Gleitkommamultiplikation und -division benutzen als Versorgungszellen die Register $R_1$ und $R_3$.

Wie bei der Addition und Subtraktion sind auch hier vier Varianten möglich. Sie können wie folgt angesprungen werden:

| | Ansprungzelle |
|---|---|
| (a) Gleitkommamultiplikation mit Speicherung | 7547 |
| (b) Gleitkommamultiplikation ohne Speicherung | 7546 |
| (c) Gleitkommadivision mit Speicherung | 7545 |
| (d) Gleitkommadivision ohne Speicherung | 7544 |

Bei den vier Varianten wird gebildet:

(a) $\langle R_1 \rangle := \langle Ac \rangle := \langle R_1 \rangle \times \langle R_3 \rangle$ 7547

(b) $\langle Ac \rangle := \langle R_1 \rangle \times \langle R_3 \rangle$ 7546

(c) $\langle R_1 \rangle := \langle Ac \rangle := \langle R_1 \rangle : \langle R_3 \rangle$ 7545 $\langle R_3 \rangle \neq 0$

(d) $\langle Ac \rangle := \langle R_1 \rangle : \langle R_3 \rangle$ 7544

Die Ausgangswerte der Register $R_1$ und $R_3$ bleiben in keinem Fall erhalten.

Eine Wertebereichsüberschreitung wird durch das Leuchten von "FÜ" und "US" angezeigt.

In der Adressenanzeige wird bei der

- Gleitkommamultiplikation die Adresse 7546
- Gleitkommadivision die Adresse 7544

angegeben.

Die Rücksprungadressen sind für alle vier Varianten in der Zelle 7766 gespeichert.

## 3.2. Festkommamultiplikation, Festkommadivision

Bei den Festkommaoperationen wird berechnet

$$c = a \cdot b \cdot 10^{-q} \quad \text{bzw.} \quad c = \frac{a}{b} \cdot 10^{q}$$

Die Größen a, b, c sind Festkommazahlen, K.h. und q ist ein Korrekturfaktor. Sowohl bei der Multiplikation als auch bei der Division ist dieser Korrekturfaktor anzugeben. Für ihn gilt:

$0 \leq q \leq 7$ ; q - ganzzahlig

Der Korrekturfaktor steht in den unbesetzten Bits $z_0\ z_1\ z_2$ des Befehls 754x 025. (Sprung mit Rückkehrabsicht zum Rechenprogramm). Mit Hilfe des Faktors q kann bei der Multiplikation und Division eine gedachte Kommastellung realisiert werden. Bei der Division hängt die Genauigkeit des Ergebnisses von q ab.

Die vier möglichen Varianten sind wie folgt anzuspringen:

| | Ansprungzelle |
|---|---|
| (a) Festkommamultiplikation mit Speicherung | 7543 |
| (b) Festkommamultiplikation ohne Speicherung | 7542 |
| (c) Festkommadivision mit Speicherung | 7541 |
| (d) Festkommadivision ohne Speicherung | 7540 |

Bei den Varianten wird gebildet:

Kommastellen beachten !

(a) $\langle R_1 \rangle := \langle Ac \rangle := \langle R_1 \rangle \times \langle R_3 \rangle \times 10^{-q}$

(b) $\langle Ac \rangle := \langle R_1 \rangle \times \langle R_3 \rangle \times 10^{-q}$

(c) $\langle R_1 \rangle := \langle Ac \rangle := \langle R_1 \rangle : \langle R_3 \rangle \times 10^{q}$

(d) $\langle Ac \rangle := \langle R_1 \rangle : \langle R_3 \rangle \times 10^{q}$

3,36 : 1,12
= 336 : 112
= 3 , soll Komma erscheinen, x 10?

Im Fall (a) und (c) bleibt der Inhalt des Registers $R_2$ erhalten, während in $R_1$ das Ergebnis steht.

Im Fall (b) und (d) werden die Inhalte der Register $R_1$ und $R_2$ nicht verändert.

Der Zahlenbereich ist bei beiden Operationen durch

$$|z| \leq 2^{31} - 1$$

beschränkt.

Eine Wertebereichsüberschreitung wird durch das Leuchten von "FÜ" und "US" angezeigt.

In der Befehlsregisteranzeige werden

- bei der Multiplikation die Adresse 7542
- bei der Division die Adresse 7540

angegeben.

Die Rücksprungadressen sind bei den Varianten ohne Speicherung in der Zelle 7741 und bei den Varianten mit Speicherung in der Zelle 7742 gespeichert. Bei der Multiplikation und Division mit Speicherung steht in der Zelle 7741 der Befehl 7773 240. (Transport nach Zelle 7773).

## 4. Ausgabe

Da beim Cellatron C 8205 nur die Ausgabe eines einzigen Zeichens verdrahtet ist, wurden für die Ausgabe von ganzen Worten Standardprogramme geschaffen.

Dabei kann die auszugebende Information verschieden interpretiert werden.

Alle Ausgabeunterprogramme sind - um die Programmierung zu erleichtern - über feste Ansprungzellen erreichbar. Außer für die im Minimalprogramm enthaltenen Ausgabeprogramme befinden sich diese Zellen auf der Bahn 7540. Es werden die Zellen 7554 bis 7563 verwendet.

Als Versorgungszelle wird bei allen Ausgabeunterprogrammen das Register $R_1$ benutzt. Der Aufruf des Unterprogramms braucht nicht direkt nach der Speicherung des auszugebenden Wortes nach $R_1$ erfolgen, wenn man beachtet, daß der Inhalt dieses Registers inzwischen nicht zerstört wird.

Der Inhalt des Akkumulators beim Rücksprung ist belanglos. Durch die Abarbeitung eines Ausgabeprogrammes wird der Inhalt von $R_1$ nicht verändert.

In den Ausgabeunterprogrammen sind keine Anwahlbefehle berücksichtigt.

### 4.1. Ausgabeprogramme des Minimalprogramms

In das Minimalprogramm wurden drei häufig benutzte Programme zum Druck von Adressen, Befehlen und Triaden aufgenommen. Diese Druckprogramme sind als Unterprogramme verwendbar. Sie können auf den folgenden Zellen angesprungen werden:

| | Ansprungzelle |
|---|---|
| UP Adressendruck | 100 |
| UP Befehlsdruck | 53 |
| UP Triadendruck | 276 |

Bsp.: <$R_1$> = 1234 5670 240

UP. 100: 5670
UP. 53: 5670 240
UP. 276: 1234 5670 240

## 4.1.1. Befehls- und Adressendruck

### Adressendruck

Beim Adressendruck werden nur die Wortstellen $z_{12}$ bis $z_{23}$ des Maschinenwortes gedruckt. Diese 12 Dualstellen werden mit 4 Oktalziffern ausgegeben. Je drei duale Wortstellen werden zu einer Oktalziffer zusammengefaßt.

Beispiel: 0000 kleinste mögliche Adresse
7777 größte mögliche Adresse

### Befehlsdruck

Es werden bei diesem Druck die Wortstellen $z_o$ bis $z_{11}$ nicht berücksichtigt, da diese für die interne Befehlsinterpretation keine Bedeutung haben.

Die Befehlsadresse wird mit 4, der Operationsteil mit 3 Oktalziffern gedruckt. Zwischen dem Druck der Adresse und des Operationsteils wird ein Leerschritt ausgeführt.

Beispiel: 0172 211
7645 442

## 4.1.2. Triadendruck

### (Ausgabe von Oktalzahlen)

Die Ausgabe des internen Wortes erfolgt hierbei in Form von 11 Oktalziffern. Nach dem Druck von jeweils 4 Oktalziffern wird ein Leerschritt ausgeführt. Damit erhält das Druckbild die Gestalt:

$$a_o a_1 a_2 a_3 \; a_4 a_5 a_6 a_7 \; a_8 a_9 a_{10}$$

Beispiel: 7 0 2 6 4 1 3 5 0 7 1

Ist in dem internen Maschinenwort das erste Bit $z_o$ besetzt, so ist die erste ausgedruckte Ziffer $a_o \geq 4$.

## 4.2. Gleitkommaausgabe

Bei den Programmen zur Gleitkommaausgabe wird der Inhalt des Registers $R_1$ als Zahl in Gleitkommadarstellung interpretiert.

### 4.2.1. Ausgabe in Gleitkommadarstellung

Mit Hilfe dieses Druckprogramms wird der Inhalt von $R_1$ als Zahl in Gleitkommadarstellung ausgegeben.

Das Druckbild umfaßt 11 Zeichen:

- das Mantissenvorzeichen (bei positivem Vorzeichen wird ein Leerschritt ausgeführt)
- 7 dezimale Mantissenziffern
- Exponentenvorzeichen (bei positivem Vorzeichen wird ein Leerschritt ausgeführt)
- 2 Ziffern für den Dezimalexponenten

$$\overset{(+)}{-}\ m_1m_2m_3m_4m_5m_6m_7\ \overset{(+)}{-}e_1e_2$$

Bsp.: intern −12,83
Druck: −1283000 0
= −0,1283 · 10²

Der Ausgabezahlenbereich stimmt mit dem internen überein. Gedruckt wird der echte, d.h. nicht erhöhte Dezimalexponent. Er kann maximal den Wert $e_1e_2 = 10$ annehmen.

Der Gleitkommanull (intern alle Bits gleich Null) ist eindeutig die Ausgabeschreibweise

0000000-09 ← Kleinster negativer Exponent

zugeordnet.

Ansprungzelle: 7562

### 4.2.2. Ausgabe in Dezimalschreibweise

Durch dieses Ausgabeprogramm wird das als Gleitkommazahl interpretierte interne Maschinenwort nach der Rückkonvertierung als Zahl in Dezimalschreibweise gedruckt.

Von dem Programm gibt es zwei Ausführungen. Sie gestatten die Ausgabe mit und ohne Runden. Die beiden Varianten können über die Zellen 7560 und 7561 er-

reicht werden.

| | Ansprungzelle |
|---|---|
| (a) Ausgabe einer Gleitkommazahl in Dezimalschreibweise ohne Runden | 7560 |
| (b) Ausgabe einer Gleitkommazahl in Dezimalschreibweise mit Runden (nach üblicher Vorschrift) | 7561 |

Die Anzahl der Zeichen, die gedruckt werden, ist abhängig von zwei Parametern. Beim Ansprung der beiden Druckprogramme ist in den unbesetzten Bits des Befehls "Sprung mit Rückkehrabsicht nach 7560 bzw. 7561" anzugeben, wieviel Stellen vor und nach dem Komma ausgegeben werden sollen. Der Parameter m, der die Anzahl der Stellen vor dem Komma festlegt, steht in den Bits $z_4$ $z_5$ $z_6$ $z_7$ des Einsprungbefehls. Der Parameter n bestimmt die Anzahl der Stellen nach dem Komma. Er ist in den Bits $z_8$ $z_9$ $z_{10}$ $z_{11}$ anzugeben. Für beide Parameter gilt:

$$0 \leq m \text{ u. } n \leq 15$$

Wenn die zu druckende Zahl mehr Stellen vor dem Komma besitzt, als m angibt, wird "ZG" (Zahl zu groß) ausgegeben.

Ein positives Vorzeichen wird nicht gedruckt, während ein negatives Vorzeichen hinter der Zahl steht.

Der Ausgabezahlenbereich stimmt wie bei der Ausgabe in Gleitkommadarstellung mit dem internen Zahlenbereich überein.

Beispiel: Es ist die Zahl -547,13 auszugeben.
Für die Parameter gilt: m = 3, n = 2
Druckbild: 547,13-
Gilt für die Parameter m = 5, n = 3, so hat das Druckbild die Gestalt:
xx547,130-
(vor dem Druck der 1. Ziffer (5) werden zwei Leerschritte ausgeführt, da m = 5 ist.)
Ist m = 1, n = 2 angegeben, so wird "ZG" gedruckt.

Bsp.! intern: 746,385
m = 3, n = 2
Druck: 746,38 (o.R.)
746,39 (m.R.)

$z_0 z_1 z_2 \mid z_3 z_4 z_5 \mid z_6 z_7 z_8 \mid z_9 z_{10} z_{11} z_{12} \ldots$ ($z_4$–$z_7$: m; $z_8$–$z_{11}$: n)

## 4.3. Festkommaausgabe

Bei den Programmen zur Ausgabe von Festkommazahlen wird das interne Wort als Dualzahl interpretiert und in eine Dezimalzahl umgewandelt. Der Druck erfolgt in Dezimalschreibweise.

In Abhängigkeit von der Druckaufbereitung unterscheidet man bei den Festkommaausgabeprogrammen zwei Varianten, die man als Variante I und Variante II bezeichnet. Bei der Variante I sind zwei Parameter notwendig, um die Anzahl der Stellen vor und nach dem Komma festzulegen. Außerdem setzt das Programm voraus, daß die Zahl vor der Ausgabe kommagerecht aufbereitet wurde. (Kommastellung wird vom Programmierer vorbereitet)

Diese spezielle Aufbereitung der Zahl wird bei Variante II nicht gefordert. Dafür sind drei Parameter anzugeben. Zwei Parameter legen die Anzahl der Stellen vor und nach dem Komma fest. Der dritte Parameter gibt an, welche Ziffer der Dezimalzahl zuerst zu drucken ist.

Die Parameter bestimmen auch die Anzahl der zu druckenden Zeichen. Das negative Vorzeichen wird am Ende einer Zahl ausgegeben. Bei einem positiven Vorzeichen wird ein Leerschritt ausgeführt.

Der Ausgabezahlenbereich ist gleich dem internen, d.h. es können Zahlen aus dem Bereich:

$$|z| \leq 2^{31} - 1 = 2\ 147\ 483\ 647$$

gedruckt werden.

### 4.3.1. Variante I

Dieses Ausgabeprogramm kann auf der Zelle 7557 angesprungen werden.

Im Einsprungbefehl werden in den Bits $z_4$ $z_5$ $z_6$ $z_7$ die Stellen vor dem Komma (m) und in den Bits $z_8$ $z_9$ $z_{10}$ $z_{11}$ die Stellen nach dem Komma (n) angegeben. Für die Parameter gilt:

$$0 \leq m \text{ u. } n \leq 15$$

Vor dem Ansprung des Druckprogramms ist die auszugebende Zahl kommagerecht aufzubereiten. Der Begriff der kommagerechten Aufbereitung soll an einem Beispiel erläutert werden. Die maximal zehn ausgebbaren Dezimalziffern seien wie folgt durchnumeriert:

$$z_9 z_8 z_7 z_6 z_5 z_4 z_3 z_2 z_1 z_0$$

! m = 10 - Stellenza[hl] hinter de[m]

Der Parameter m bestimmt, wieviel Ziffern von $z_9$ ab beim Druck der Stellen vor dem Komma zu berücksichtigen sind. Der Parameter n gibt an, wieviel Ziffern nach dem Druck des Kommas ausgegeben werden. Ist m = 5 und n = 2, so werden die Ziffern $z_9$ $z_8$ $z_7$ $z_6$ $z_5$ für den Druck der Stellen vor dem Komma und die Ziffern $z_4$ $z_3$ für den Druck der Stellen nach dem Komma berücksichtigt.

Damit die Zahl kommagerecht ausgegeben wird ist Voraussetzung, daß die gedachte Kommastellung der zu druckenden Zahl der durch den Parameter m festgelegten Kommastellung entspricht. Wenn z.B. die Zahl -347,15 gedruckt werden soll und für die Parameter m = 5 und n = 2 gilt, so muß intern die Zahl -34715000 aufbereitet sein. Das Druckbild hat in diesem Fall das folgende Aussehen:

xx347,15-

(Vor dem Druck der ersten Ziffer (3) werden zwei Leerschritte ausgeführt.)

Steht im Register $R_1$ die oben angegebene Festkommazahl, K.h.

-34715000

und ist m = 3, n = 2, so wird

3,47-

Runden bei Anwendung am Bsp. prüfen. - Text unvollständig

gedruckt.

Wäre m = 2, n = 1, so würde

0,3-

ausgegeben.

### 4.3.2. Variante II

Von diesem Ausgabeprogramm existieren zwei Ausführungen. Sie können über die Ansprungzellen 7555 und 7556 erreicht werden.

| | Ansprungzelle |
|---|---|
| (a) Ausgabe von Festkommazahlen, K.h. ohne Runden | 7555 |
| (b) Ausgabe von Festkommazahlen, K.h. mit Runden | 7556 |

Im Einsprungbefehl sind drei Parameter anzugeben:

1. i - erste zu druckende Ziffer in den Bits: $z_0 z_1 z_2 z_3$
2. m - Anzahl der Stellen vor dem Komma in den Bits: $z_4 z_5 z_6 z_7$
3. n - Anzahl der Stellen nach dem Komma in den Bits: $z_8 z_9 z_{10} z_{11}$

Wenn die zehn Dezimalziffern wieder von 0 bis 9 durchnumeriert werden, so gibt der Parameter i ($0 \leq i \leq 9$) die höchste Ziffer an, die bei der Ausgabe berücksichtigt wird.

Beispiel: Für die Parameter m und n gilt: m=5, n=2. Wenn die interne Festkommazahl, K.h. 34715 in der Form 347,15 ausgegeben werden soll, muß der Parameter i = 6 sein.

Ist die Zahl intern mit 34715000 angegeben, muß für den Parameter i = 9 gelten (m = 5, n = 2). In beiden Fällen hat das Druckbild die Gestalt

xx347,15

(Vor dem Druck der ersten Ziffer (3) wird zweimal Zwischenraum ausgegeben.)

Wenn die auszugebende Zahl größer ist als der Parameter ~~i angibt~~, ~~d.h.,~~ wenn Dezimalziffern vor der durch i bestimmten Ziffer ~~ungleich~~ Null sind, so wird

"ZG" (Zahl zu groß)

gedruckt.

Ist in $R_1$ die Zahl

$$-34715000$$

angegeben und ist $i \leq 6$, so erfolgt für beliebiges m und n der Druck von "ZG".

4.4. Tetradenausgabe

Dieses Ausgabeprogramm interpretiert das interne Maschinenwort als Zahl in Tetradendarstellung. Das Programm konvertiert diese Zahl in eine Festkommazahl, K.h. Der Druck der Festkommazahl erfolgt wie bei Variante II.

Das Programm kann auf der Zelle 7554 angesprungen werden. Im Befehl "Sprung mit Rückkehrabsicht" sind wie bei Variante II die drei Parameter

1. i=erste zu druckende Ziffer in den Bits: $z_0$ $z_1$ $z_2$ $z_3$ ( $0 \leq i \leq 7$.
2. m=Anzahl der Stellen vor dem Komma in den Bits: $z_4$ $z_5$ $z_6$ $z_7$
3. n=Anzahl der Stellen nach dem Komma in den Bits: $z_8$ $z_9$ $z_{10}$ $z_{11}$

anzugeben.

Der Ausgabezahlenbereich ist durch

$$z \leq 10^8 - 1$$

beschränkt, d.h. er stimmt mit dem internen Zahlenbereich überein.

Auch bei diesem Programm wird "ZG" gedruckt, wenn die auszugebende Zahl größer ist als der Parameter i festlegt.

Tritt eine Pseudotetrade auf, d.h. einer Tetrade

entspricht eine Zahl z mit $10 \leq z \leq 15$, so wird „P" "P" (Pseudotetrade) ausgeschrieben.

## 4.5. Triadenausgabe auf Lochband

Mit Hilfe dieses Programmes wird der Inhalt des Registers $R_1$ in einer Form gestanzt, in der er wieder vom Rechner eingelesen werden kann. Vor dem Stanzen der 11 Triaden werden einmal das Zeichen Zwischenraum "ZwR" und die Eingabekennzeichen $K_1 = 0$ und $K_2 = 0$ ausgegeben. ZR 00 xxxx xxxx xxx

Ein Wort wird in der 5- und in der 8-Kanal-Variante durch die Wortmarke Punkt "." abgeschlossen.

Ausprägung über Hilfsprogramm mit „D" u. einem weiteren Kennzeichen